

MotionWeaver：プロフェッショナル技術仕様書 (v10.0)

～ 空間解析エンジン、リターゲット論理、および基幹データ構造の定義 ～

最終更新：2026/04/21

本ドキュメントは、MotionWeaver（空間解析エンジン）の全技術仕様をまとめた技術仕様書である。アニメーターがその内部論理を理解することで、3Dの幾何学的なデータを2Dアニメのデフォルメ表現へと適切に変換し、様々な3D資産を自身の表現として活用することを目的とする。

1. 空間解析エンジンの基本原理

MotionWeaverは、3D空間における「骨肉（ボーン）の回転データ（BVH）」を、2.5Dアニメーション環境（Moho）に適合するように、幾何学的な座標変換を行う高精度モーション流し込みエンジンである。

1.1 オイラー角の分解と再構成

3Dソフト（PoserやBlender等）で一般的に使われる **XYZ回転順序** を、Moho固有のX-Y-Z座標系へと動的に分解・再計算する。

- **Conversion Law（変換の法則）**：3Dの奥行き（Z座標）を2.5Dの奥行き（Z値およびZ順序）へ変換する際、物理的な軸の入れ替え（X軸とZ軸の交換など）をプロファイルに基づいて自動実行する。

1.2 リターゲットング：ユニバーサル・ハブ

異なるボーン階層を持つモデル間でのモーション移送（流し込み）を実現する。

- **型（Template）による補正**：ターゲットとなるBVHを読み込むことで、ボーンの初期角度（レストポーズ）の差異を自動計算し、偏差のみを抽出・適用する。

1.3 回転の連続性保護 (Ambiguity Resolution)

3D回転データの計算において発生する多義性（同じ姿勢に対して複数の数値が存在する問題）を解決し、アニメーションの「跳ね」を抑制する。

- **Euler Filter**：前フレームの回転値を参照し、最短距離の角度を選択することで、激しいアクション時でもボーンが捻じ切れることのない滑らかな連続性を保証する。

1.4 空間投影と透視定数 (Spatial Projection)

3D空間の奥行き情報を、2.5Dキャンバス上の「スケール」と「座標」へと忠実に変換するための幾何学的定数。

- **VIZ_ZOOM_FACTOR (3.732...)**：カメラの画角とキャンバス幅を繋ぐ固有の倍率定数。これにより、3D空間上の1cmの動きが、2.5Dキャンバス上の正確なピクセル量として再現される。
- **Camera Z (透視強度)**：3D空間の奥行き情報を、2.5Dキャンバス上の「スケール変化」へと変換するための距離定数である。

- **デフォルト動作:** 変換実行時、または CUI モードにおいて引数による明示的な指定がない場合、`templates.json` 内の `motion_perspective_default_z`（標準値 8.0）が参照される。
- **工学的役割:** この値はパースの「圧縮率」を決定する。値が小さいほどパースが強くなり、キャラクターがカメラに近づいた際の巨大化が顕著になる。逆に値を大きくすると、望遠レンズや平行投影に近い、サイズ変化の少ない安定した出力が得られる。
- **データソースへの適合:** TDPT 等の奥行き移動が激しいソースを扱う場合でも、この値を適切に設定することで、イラストとして破綻のない自然な遠近感を維持することが可能となる。

2. 基幹ファイル物理仕様と情報の蓄積 (The Configuration & Growth)

Weaverの汎用性は、プログラム（ロジック）とデータ（設定定義）を完全に分離した「設定ベース」の設計にあります。

2.1 bvh_knowledge.json (統計的マッピング推論システム)

ユーザーが過去に行ったマッピング（流し込み設定）は、`bvh_knowledge.json` に統計データとして自動蓄積される。

- **推論アルゴリズム:** 未知のBVHを読み込んだ際、過去の成功例からもっとも確率の高い流し込み対応（例：`Pelvis` → `Hips`）を自動的に提案する。
- **辞書の最適化:** 画面右上の「辞書(知識ベース)を更新」ボタンにより、現在の作業ディレクトリ内の全プロファイルを再スキャンし、推論精度を最新状態へ最適化する。

2.2 synonyms.json (表記揺れ吸収)

`Hips`、`Pelvis`、`Waist` といった、ソフトウェアごとに異なる部位名称を同一視するための「シノニム（同義語）辞書」を内蔵している。これにより、世界中のBVH素材を規格の壁を越えて透過的に扱うことが可能となる。

2.3 templates.json の探索プライオリティ (3-tier Search)

設定ファイル `templates.json` は、環境の柔軟性を守るために以下の優先順位（3階層）で自動探索される。

1. **Local (PSDフォルダ直下):** 開いているPSDと同じ場所に設定ファイルを置くことで、特定キャラクター専用のルール（ボーン構成等）を最優先で適用できる。
2. **Pointer (外部ポインタ指定):** `templates_path.txt` に記述された絶対パスを参照。チーム制作等でサーバー上の共通設定を各クライアントが参照する場合に使用される。
3. **System (アプリフォルダ):** `MotionWeaver.py` と同階層にあるデフォルト設定。上記のいずれも見つからない場合の最終的なフォールバック（Base Configuration）として機能する。

2.4 templates.json：マスターカタログと骨格の定義 (Master Catalog)

あらゆるプロファイルの「絶対基準」を定義したシステムのインデックスであり、PSDリギングへの設計図でもある。

- **骨格トポロジーと標準姿勢:** `Human_V2`（二足）や `Quadruped`（四足）などのボーン親子関係、および変換の起点となる「レストポーズ」を規定。これがMotionWeaverにおける視覚的なスケール感や、流し込み時の偏差計算の「ゼロ地点」を決定する。

- **自動リギングへの継承**: ここに定義された階層構造は、そのまま RigWeaver における Moho ボーン構築の設計図（リファレンス）としてバトンリレーされる。

2.5 言語リソースの拡張 (Internationalization)

Weaverシステムは、グローバルな制作環境を支えるため、UI上の全文字列が外部のJSONリソースとして完全に独立定義されている。

- **lang.json の自動スキャン**: アプリケーション起動時に実行ディレクトリ内の **lang*.json** を一括スキャンし、ファイル内の **_language_name** キーの値を元に、多言語切り替えメニューを動的に構築する。
- **ローカライズの論理**: 既存の **langja.json** を複製し、値を翻訳して別名（例：**langcn.json**）で保存するだけで、システムの工学的な論理を汚すことなく、任意の言語環境を構築することが可能である。

3. 運用フローとデータ管理 (Logical Management)

MotionWeaverは、単体での直感的なUI操作に加え、Weaverシリーズ全体の自動化を支えるための高度なデータ出力・外部連携仕様を備えている。

3.1 自動発注プロトコル (Argument Protocol)

MotionWeaverは、OSのコマンドラインからの「非対面実行（Silent Execution）」をサポートしており、Moho（RigWeaver）によって生成される **RUN.bat** は、このプロトコルに基づいた「自動発注書」として機能する。

引数	物理的役割	詳細
-src	ソースBVHパス	流し込み元の「演技」データを指定。
-tgt	ターゲットBVHパス	出力の規準となる「型」を指定。
-prof	プロファイルパス	紐付け（対応辞書）ファイル（.json）を指定。
-out	出力CSVパス	2.5D空間座標データ（_3d.csv）の保存先を指定。
-rig	出力RIGパス	骨格階層データ（_rig.csv）の保存先を指定。
-view	視点パラメータ	変換時のカメラ角度（x,y,z,rx,ry,rz）を定義。
-mode	処理モード	開始フレームの論理（1 または 2）を決定。

3.2 出力データの物理スキーマと役割 (Data Specs)

流し込み実行後、指定のディレクトリに以下のCSVファイルが「データのバトン」として生成される。

1. ***_3d.csv (座標・回転データ)**: キャラクターの空間座標（X, Y, Z）と、Root（腰）の回転角（rX, rY, rZ）を格納。これが Moho 上のボーンの「動き」と「奥行き」の源泉となる。
2. ***_rig.csv (骨格定義データ)**: PSDから判別された全ボーンの「親子関係（階層構造）」を記述。RigWeaver はこのファイルを参照して、Moho 上に正確なボーン階層を自動構築する。

3.3 視点アングル制御 (Virtual Camera)

3Dモーションを「正面・横・煽り・俯瞰」など、2Dイラストのパスに合わせた最適なアングルで切り出すための空間補正ロジック。

- **空間投影の最適化:** カメラの回転オフセットを計算に含めることで、実際の3D空間上の動きを、2.5Dキャンバス上での「最も見栄えの良い投影」へと動的に変換する。特に関節の重なりやパスの圧縮を、イラストの意図に合わせるための演出的な座標適合プロセスである。

3.4 回転スパイクの自動抑制 (Spike Suppression)

3Dデータ特有のノイズや、計算の多義性によって発生するボーンの急激な「跳ね (Spike)」を物理的に抑制する内部フィルタ。

- **浄化の仕様 (Filter):** フレーム間の回転変化量が内部閾値を超えた場合、システムはそれを「物理的に不可能なノイズ」と判定し、前後の値から滑らかに補間・減衰させる。これにより、2Dアニメらしい落ち着いた挙動が保証される。

4. 高度な診断とモード選択の論理 (Logical Diagnostics)

4.1 骨の強奪 (Usurping) と競合解決

1つのターゲットボーンに対して2つ目のソースボーンを割り当てようとした際、システムは「**強奪 (Usurping)**」の警告を発し、物理的な不整合を未然に防止する。

- **論理的背景:** 1つの筋肉に2つの脳からの指令が届くような矛盾を防ぐための、数学的整合性の保護機能である。

4.2 処理プロセス (Mode 1 と Mode 2) の論理的選択

アニメーション構築の成否を分ける、開始フレーム処理の論理的な使い分け基準である。

1. **Mode 1 (標準ポーズ同調 / イラストベース):** あなたが描いた「イラストのポーズ」をそのまま計算の起点として採用する。ボーンの初期角度が自然に保たれるため、通常のセットアップではこのモードが最も直感的かつ高精度な結果をもたらす。
2. **Mode 2 (基準姿勢抽出 / トリム安全装置):** イラストのポーズが極端な場合や、BVHの長い尺から一部を切り出す「トリム (Trim)」運用時に必須となるレスキューモード。
 - **インジェクション論理:** トリムによって切り捨てられたはずの「BVH本来の第1フレーム (基準姿勢)」を、強制的に出力データの第0フレームへと抽出・復元する。
 - **実務上のメリット:** トリム開始地点が激しいアクション (屈む、腕を上げる等) であっても、Rigging エンジンはこの復元された第0フレームを「Rest Pose (計算の原点)」として参照する。これにより、ポーズの乖離によるメッシュの崩壊 (怪物化) を論理的に回避し、安定したリギングの定着を実現する。

4.3 SOS プロトコル: エンジニアリング支援

未知のBVH構造において流し込みが破綻した場合、当てずっぽうな調整は厳禁である。

- **提供データ:** `MotionWeaver.log` の全文。ここには数学的な演算過程と、どのボーンで計算異常 (NaN 等) が発生したかが詳細に記録されている。
- **解決策:** ログに基づき、プロファイル内の `initial_angles` を微調整するだけで、大抵の捻じれは解決可能である。

4.4 例外ハンドリング：中和（Neutralize）と削除（None）の論理

UI上のマッピングコマンドは、データの物理的な連続性を意図的に「断ち切る」または「固定する」ための強力な論理介入である。

None (削除) の論理

ソース側の joint を計算プロセスから完全にパージ（除外）し、出力データ（CSV）への書き出しを停止する。

- **論理的実体:** データフローの遮断。
- **運用上の役割:** 2Dリグ側に存在しない3D特有の補助骨や、リギングの計算に悪影響を及ぼすノイズ源を「存在しないもの」として扱う。これにより、RigWeaver における構築エラーを回避し、データの整合性を保つ。

Neutralize (中和 / 固定) の論理

対象ボーンの全フレームにわたる動的な変化を破棄し、「BVHソースの第1フレーム（Frame 0）」の数値のみを強制適用してロックする。

- **論理前実体:** 静的ポーズ（Static Pose）へのクランプ。
- **中和の3形態（工学的選択）:**
 1. **中和 (全):** 回転と移動の両方を単一値で固定。Rootボーンの移動を抑制してその場で演技させる場合や、特定の主要骨を不動にしたい場合に使用する。
 2. **中和 (回転):** 移動（Position）のデータは活かしつつ、回転（Rotation）のみをイラストの初期角度に固定する。ボーンの「向き」をイラストの都合で一定に保ちたい場合に有効。
 3. **中和 (移動):** 回転（Rotation）のデータは活かしつつ、移動（Position）のみを固定する。ボーンのピボット位置を動かさず、その場での回転の演技のみを反映させたい場合に最適である。

【現場の知恵】余剰ボーンの「中和」による論理的封印

ターゲットリグに存在しない末端の骨（例：指の第一関節）がBVHソースに含まれる場合、単に「名前維持」で流すのではなく、明示的に「中和」を適用することがプロの作法である。

- **データの浄化（Data Hygiene）:** 「名前維持」では、ターゲットに存在しない骨の激しいモーションデータがCSVにそのまま出力される。「中和」を適用することで、これらの不要な列の値を「静止（Frame 0の値）」に書き換え、CSVデータを工学的にクリーンな状態に保つことができる。
- **将来の拡張性保護（Future-proofing）:** 後にリグを改修してその骨を追加した際、過去に生成したCSVを流し込んでも、当時のBVHに含まれていた予期せぬノイズや歪んだポーズが突然反映されるリスクを回避できる。
- **構造の安定性:** 「削除（None）」とは異なり、CSVの列構造そのものは維持されるため、Weaverシリーズ全体のデータ整合性を崩すことなく、安全にモーションのみを「論理的に封印」することが可能となる。

5. 結論：柔軟な表現を支える変換論理

MotionWeaverは、3Dデータの数学的正当性を2Dアニメーションの論理へと橋渡しし、効率的な制作パイプラインを構築するための変換エンジンである。データの仕組みを理解し、ツールを適切に運用することで、

様々な3D資産を効率的に活用し、アニメーション制作の幅を広げることができるだろう。

6. UI統合リファレンス (GUI Logic Mapping)

本章では、GUI上の各設定項目が内部のどの論理セクションと紐付いているかを定義する。

6.1 ファイル操作と知識ベース (File & Knowledge)

UI項目	物理機能・役割	関連論理 (本仕様書リンク)
変換元 BVH	流し込み元のモーションデータ（ボーンの回転情報）を指定。	第1章：空間解析の入力
ターゲット/プロファイル	モーションを適用する「型」または既成の設定ファイルを指定。	第2.4節：マスターカタログ
PSD 静止リグ生成専用	PSDからボーン構造を抽出する専用セクション。BVHフローとは独立して動作。	第2.4節：リギング設計図
辞書を更新	カレントフォルダの情報を再スキャンし統計データを最新化。	第2.1節：統計的推論
言語(Language)	lang*.json を切り替え、すべてのUI表示語を置換。	第2.5節：多言語対応

6.2 マッピング編集コマンド (Mapping Controls)

UI項目	物理機能・役割	内部論理・処理
< 割り当て	ソース joint をターゲット bone に紐付ける。	第2.1節：マッピング推論
< 名前維持	ソース名のまま変換。BVH構造が同一の場合に有効。	リターゲットの最小化
< 中和 (全/回/移)	動きを破棄し特定値をロック。Root固定や末端演技に使用。	第4.4節：中和の論理
< None 削除	対象ボーンを計算プロセスからパージしデータ出力を遮断。	第4.4節：削除の論理
L -> R 自動入力	左右対称ボーンを自動補完。特殊タグや L/R 置換を高精度で実行。	ミラーリング論理の強化

6.3 モーション設定ダイアログ (Motion Settings)

UI項目	物理機能・役割	内部論理・処理
開始フレーム (Mode 1)	モーションデータの開始フレームをそのまま起点とする。	第4.2節：標準ポーズ同調

UI項目	物理機能・役割	内部論理・処理
開始フレーム (Mode 2)	第1フレーム（基準姿勢）を抽出し先頭に強制挿入。	第4.2節：トリム安全装置
スパイク除去 (度数)	フレーム間の急激な回転変化を検知し減衰・補間。	第3.4節：スパイク抑制
トリミング (範囲指定)	指定された時間軸のみを 0.0s 起点のデータとして再構築。	空間・時間軸の再定義

6.4 視点設定と診断パネル (Viewpoint & Diagnostics)

UI項目	物理機能・役割	内部論理・処理
視点調整...	Root ボーンに対し全フレーム共通のオフセットを付与。	第3.3節：Virtual Camera
3D プレビュー	2カラム右側に常時表示。空間上の姿勢をリアルタイムに視覚化。	第1.4節：空間投影
Diagnostics (口グ)	2カラム右下に常時表示。整合性の検証と変換履歴を記録。	第4章：高度な診断

第7章：templates.json 主要パラメータ対照リファレンス (MotionWeaver専用)

MotionWeaver は、3D モーションの解析、スケール変換、および 2D 向けデータ書き出しの際、`templates.json` の以下の項目を参照します。

パラメータ名	工学的役割	設定例
<code>hip_offset_axis</code>	腰の垂直軸定義: BVH データ内の Hip ボーンの上下移動に対応する座標軸。	"Y" または "Z"
<code>mismatch_indicators</code>	スケール誤差検知: モデルのサイズ比較に使用する主要ボーン。誤差が大きい場合に警告を発する。	["Spine", "Shoulder", "Chest"]
<code>horizontal_parts</code>	水平構造定義: 四足歩行キャラ等の、水平方向に連なるボーン階層を定義する。	["Spine", "Neck", "Head"]
<code>joint_maps</code>	ボーン翻訳辞書: 入力 3D ボーン名（ソース）を 2D ボーン名（ターゲット）へ自動変換するためのシノニム。	{"Pelvis": "Hip", "Spine1": "Chest"}
<code>motion_perspective_default_z</code>	デフォルト透視定数: 変換時に Camera Z（透視強度）が指定されていない場合、またはプレビュー時に使用される標準のパース基準値。	8.0